



Big Data Analytics

Presented by: Dr Sherin El Gokhy



Module 4 – Advanced Analytics - Theory and Methods



Introduction



Analytics Lifecycle



Basic Methods



Adv. Methods



Tools



Lab

Module 4: Advanced Analytics – Theory and Methods

Part 5: Naïve Bayesian Classifiers

During this Part the following topics are covered:

- Naïve Bayesian Classifier
- Theoretical foundations of the classifier
- Use cases
- Evaluating the effectiveness of the classifier
- The Reasons to Choose (+) and Cautions (-) with the use of the classifier

Classifiers

Where in the catalog should I place this product listing?
Is this email spam?

- Classification: assign labels to objects based on the object's attributes.
- Usually supervised: training set of pre-classified examples.
- Our examples:
 - ▶ Naïve Bayesian
 - ▶ Decision Trees
 - ▶ (and Logistic Regression)

Naïve Bayesian Classifier

- The Naïve Bayesian Classifier is a probabilistic classifier based on Bayes' Law and naïve conditional independence assumptions.
- Determine the most probable class label for each object
 - ▶ Based on the observed object attributes
 - ▶ Naïvely these attributes assumed to be conditionally independent of each other
 - ▶ Example:
 - ▶ Based on the objects attributes {shape, color, weight}
 - ▶ A given object that is {spherical, yellow, < 60 grams}, may be classified (labeled) as a tennis ball
 - ▶ Even if these features depend on each other or upon the existence of the other features, a Naïve Bayesian Classifier considers all of these features independently contribute to the probability that the object is a tennis ball.
 - ▶ Class label probabilities are determined using Bayes' Law
- **Input** variables are **discrete** but there are variations to the algorithms that work with continuous variables as well
- **Output**:
 - ▶ Probability score – proportional to the true probability
 - ▶ Class label – based on the highest probability score

Naïve Bayesian Classifier - Use Cases

- Preferred method for many text classification problems.
 - ▶ Try this first; if it doesn't work, try something more complicated
- Use cases
 - ▶ Spam filtering, other text classification tasks
 - ▶ Fraud detection.....For example in auto insurance, based on a training data set with attributes (such as driver's rating, vehicle age, vehicle price, is it a claim by the policy holder, police report status, claim genuine) we can classify a new claim as genuine



Building a Training Dataset to Predict Good or Bad Credit

- Predict the credit behavior of a credit card applicant from applicant's attributes:
 - ▶ Personal status
 - ▶ Job type
 - ▶ Housing type
 - ▶ Savings amount
- These are all categorical variables and are better suited to Naïve Bayesian Classifier than to logistic regression.
- If there are multiple levels for the outcome you want to predict, then Naïve Bayesian Classifier is a better solution.

personal_status	job	housing	savings_status	credit_class
male single	skilled	own	no known savings	good
female div/dep/mar	skilled	own	<100	bad
male single	unskilled resident	own	<100	good
male single	skilled	for free	<100	good
male single	skilled	for free	<100	bad
male single	unskilled resident	for free	no known savings	good
male single	skilled	own	500<=X<1000	good
male single	high qualif/self emp/mgm	rent	<100	good
male div/sep	unskilled resident	own	>=1000	good
male mar/wid	high qualif/self emp/mgm	own	<100	bad
female div/dep/mar	skilled	rent	<100	bad
female div/dep/mar	skilled	rent	<100	bad
female div/dep/mar	skilled	own	<100	good
male single	unskilled resident	own	<100	bad
female div/dep/mar	skilled	rent	<100	good
female div/dep/mar	unskilled resident	own	100<=X<500	bad
male single	skilled	own	no known savings	good
male single	skilled	own	no known savings	good
female div/dep/mar	high qualif/self emp/mgm	for free	<100	bad
male single	skilled	own	500<=X<1000	good
male single	skilled	own	<100	good
male single	skilled	rent	500<=X<1000	good
male single	unskilled resident	rent	<100	good
male single	skilled	own	100<=X<500	good
male mar/wid	skilled	own	no known savings	good
male single	unskilled resident	own	<100	good
male mar/wid	unskilled resident	own	<100	good

Technical Description - Bayes' Law

$$P(C | A) = \frac{P(A \cap C)}{P(A)} = \frac{P(A | C)P(C)}{P(A)}$$

- C is the class label:
 - ▶ $C \in \{C_1, C_2, \dots, C_n\}$
- A is the observed object attributes
 - ▶ $A = (a_1, a_2, \dots, a_m)$
- $P(C | A)$ is the probability of C given A is observed
 - ▶ Called the conditional probability



Reverend Thomas Bayes

Technical Description - Bayes' Law

- **An example using Bayes Law:**

John flies frequently and likes to upgrade his seat to first class. He has determined that, if he checks in for his flight at least two hours early, the probability that he will get the upgrade is **.75**; otherwise, the probability that he will get the upgrade is **.35**. With his busy schedule, he checks in at least two hours before his flight only **40%** of the time. Suppose John didn't receive an upgrade on his most recent attempt. What is the probability that he arrived late?

- C = John arrives late $P(C)$ = Probability John arrives late = .6
- A = John did not receive an upgrade
- $P(A)$ = Probability John did not receive an upgrade =
$$1 - (.4 \times .75 + .6 \times .35) = 1 - .51 = .49$$

Technical Description - Bayes' Law

- **An example using Bayes Law:**
- $P(A|C)$ = Probability that John did not receive an upgrade given that he arrived late = $1 - .35 = .65$
- $P(C|A)$ = Probability that John arrived late given that he did not receive his upgrade = $P(A|C)P(C)/P(A) = (.65 \times .6)/.49 = .80$

In this simple example, C can take one of two possible values {arriving early, arriving late) and there is only one attribute which can take one of two possible values {received upgrade, did not receive upgrade}.

Apply the Naïve Assumption and Remove a Constant

- For observed attributes $A = (a_1, a_2, \dots, a_m)$, we want to compute

$$P(C_i | A) = \frac{P(a_1, a_2, \dots, a_m | C_i)P(C_i)}{P(a_1, a_2, \dots, a_m)} \quad i = 1, 2, \dots, n$$

and assign the classifier, C_i , with the largest $P(C_i | A)$

- Two simplifications to the calculations
 - Apply naïve assumption - each a_j is conditionally independent of each other, then

$$P(a_1, a_2, \dots, a_m | C_i) = P(a_1 | C_i)P(a_2 | C_i) \cdots P(a_m | C_i) = \prod_{j=1}^m P(a_j | C_i)$$

- Denominator $P(a_1, a_2, \dots, a_m)$ is a constant and can be ignored

Building a Naïve Bayesian Classifier

- Applying the two simplifications

$$P(C_i | a_1, a_2, \dots, a_m) \propto \left(\prod_{j=1}^m P(a_j | C_i) \right) P(C_i) \quad i = 1, 2, \dots, n$$

- To build a Naïve Bayesian Classifier, collect the following statistics from the training data:

- ▶ $P(C_i)$ for all the class labels.
- ▶ $P(a_j | C_i)$ for all possible a_j and C_i
- ▶ Assign the classifier label, C_i , that maximizes the value of

$$\left(\prod_{j=1}^m P(a_j | C_i) \right) P(C_i) \quad i = 1, 2, \dots, n$$

Naïve Bayesian Classifiers for the Credit Example

- Class labels: {good, bad}
 - ▶ $P(\text{good}) = 0.7$
 - ▶ $P(\text{bad}) = 0.3$
- Conditional Probabilities
 - ▶ $P(\text{own} | \text{bad}) = 0.62$
 - ▶ $P(\text{own} | \text{good}) = 0.75$
 - ▶ $P(\text{rent} | \text{bad}) = 0.23$
 - ▶ $P(\text{rent} | \text{good}) = 0.14$
 - ▶ ... and so on

Naïve Bayesian Classifier for a Particular Applicant

- Given an example of an applicant whose attributes are
 $A = \{\text{female single, owns home, self-employed, savings} > \$1000\}$
- Since $P(\text{good} | A) > P(\text{bad} | A)$, assign the applicant the label "good" credit

a_j	C_i	$P(a_j C_i)$
female single	good	0.28
female single	bad	0.36
own	good	0.75
own	bad	0.62
self emp	good	0.14
self emp	bad	0.17
savings>1K	good	0.06
savings>1K	bad	0.02

$$P(\text{good} | A) \sim (0.28 * 0.75 * 0.14 * 0.06) * 0.7 = 0.0012$$

$$P(\text{bad} | A) \sim (0.36 * 0.62 * 0.17 * 0.02) * 0.3 = 0.0002$$

Naïve Bayesian Implementation Considerations

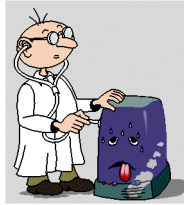
- Numerical underflow
 - ▶ Resulting from multiplying several probabilities near zero
 - ▶ Preventable by computing the logarithm of the products
- Zero probabilities due to unobserved attribute/classifier pairs
 - ▶ Resulting from rare events
 - ▶ Handled by smoothing (adjusting each probability by a small amount)
- Assign the classifier label, C_i , that maximizes the value of

$$\left(\sum_{j=1}^m \log P'(a_j | C_i) \right) + \log P(C_i)$$

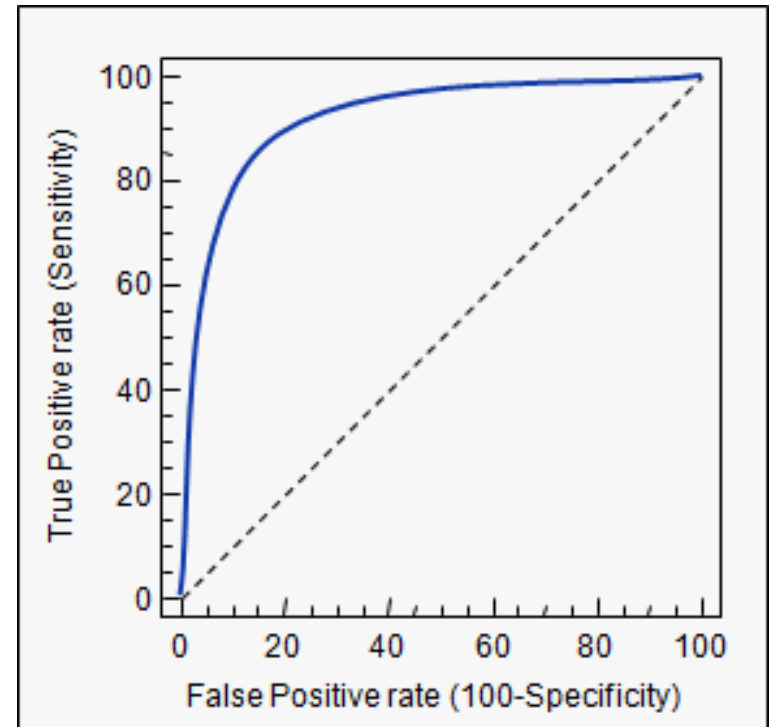
where $i = 1, 2, \dots, n$ and

P' denotes the adjusted probabilities

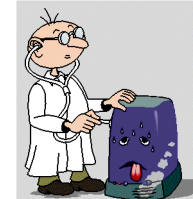
Diagnostics



- Hold-out data
 - ▶ How well does the model classify new instances?
- Cross-validation
- ROC curve/AUC



Diagnostics: Confusion Matrix



		Prediction		
Actual Class	good	bad		
	671	29	700	
bad	38	262	300	
	709	291	1000	

Annotations:

- true positives (TP) points to 671
- false negatives (FN) points to 29
- false positives (FP) points to 38
- true negatives (TN) points to 262

Overall success rate (or accuracy):

$$(TP + TN) / (TP + TN + FP + FN) = (671 + 262) / 1000 \approx 0.93$$

Recall (or TPR): $TP / (TP + FN) = 671 / (671 + 29) = 671 / 700 \approx 0.96$
what percent of positive instances did we correctly identify.

FPR: $FP / (FP + TN) = 38 / (38 + 262) = 38 / 300 \approx 0.13$
what percent of negatives we marked positive

FNR: $FN / (TP + FN) = 29 / (671 + 29) = 29 / 700 \approx 0.04$
what percent of positives we marked negative

Precision: $TP / (TP + FP) = 671 / 709 \approx 0.95$
what percent of things we marked positive really are positive

Naïve Bayesian Classifier - Reasons to Choose (+) and Cautions (-)



Reasons to Choose (+)	Cautions (-)
Handles missing values quite well	Numeric variables have to be discrete (categorized) Intervals
Robust to irrelevant variables	Sensitive to correlated variables "Double-counting"
Easy to implement	Not good for estimating probabilities Stick to class label or yes/no used for class label assignments only
Easy to score (predict) data	
Resistant to over-fitting	
Computationally efficient Handles very high dimensional problems Handles categorical variables with a lot of levels	

Check Your Knowledge



Your Thoughts?

1. Consider the following Training Data Set:

- Apply the Naïve Bayesian Classifier to this data set and compute the probability score for $P(y = 1 | X)$ for $X = (1, 0, 0)$

Show your work

Training Data Set

X1	X2	X3	Y
1	1	1	0
1	1	0	0
0	0	0	0
0	1	0	1
1	0	1	1
0	1	1	1

2. List some prominent use cases of the Naïve Bayesian Classifier.
3. What gives the Naïve Bayesian Classifier the advantage of being computationally inexpensive?
4. Why should we use log-likelihoods rather than pure probability values in the Naïve Bayesian Classifier?

Check Your Knowledge (Continued)



5. What is a confusion matrix and how it is used to evaluate the effectiveness of the model?
6. Consider the following data set with two input features temperature and season
 - What is the Naïve Bayesian assumption?
 - Is the Naïve Bayesian assumption satisfied for this problem?

Temperature	Season	Electricity Usage
-10 to 50 F	Winter	High
50 to 70 F	Winter	Low
70 to 85 F	Summer	Low
85 to 110 F	Summer	High



Introduction



Analytics Lifecycle



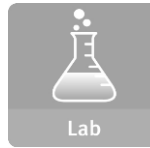
Basic Methods



Adv. Methods



Tools



Lab

Module 4: Advanced Analytics – Theory and Methods

Part 5: Naïve Bayesian Classifiers - Summary

During this Part the following topics were covered:

- Naïve Bayesian Classifier
- Theoretical foundations of the classifier
- Use cases
- Evaluating the effectiveness of the classifier
- The Reasons to Choose (+) and Cautions (-) with the use of the classifier

Lab Exercise 8: Naïve Bayesian Classifier

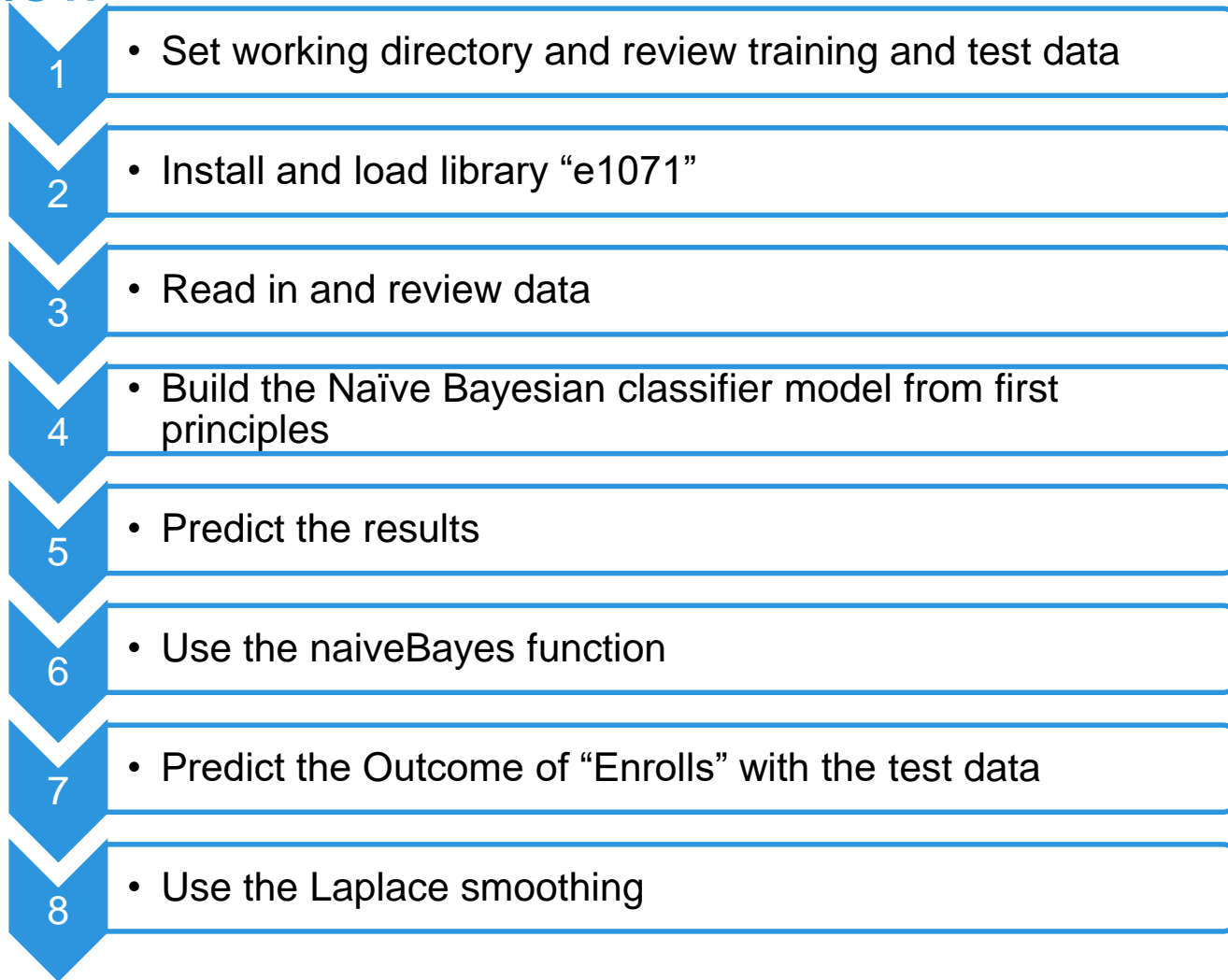


This Lab is designed to investigate and practice the Naïve Bayesian Classifier analytic technique.

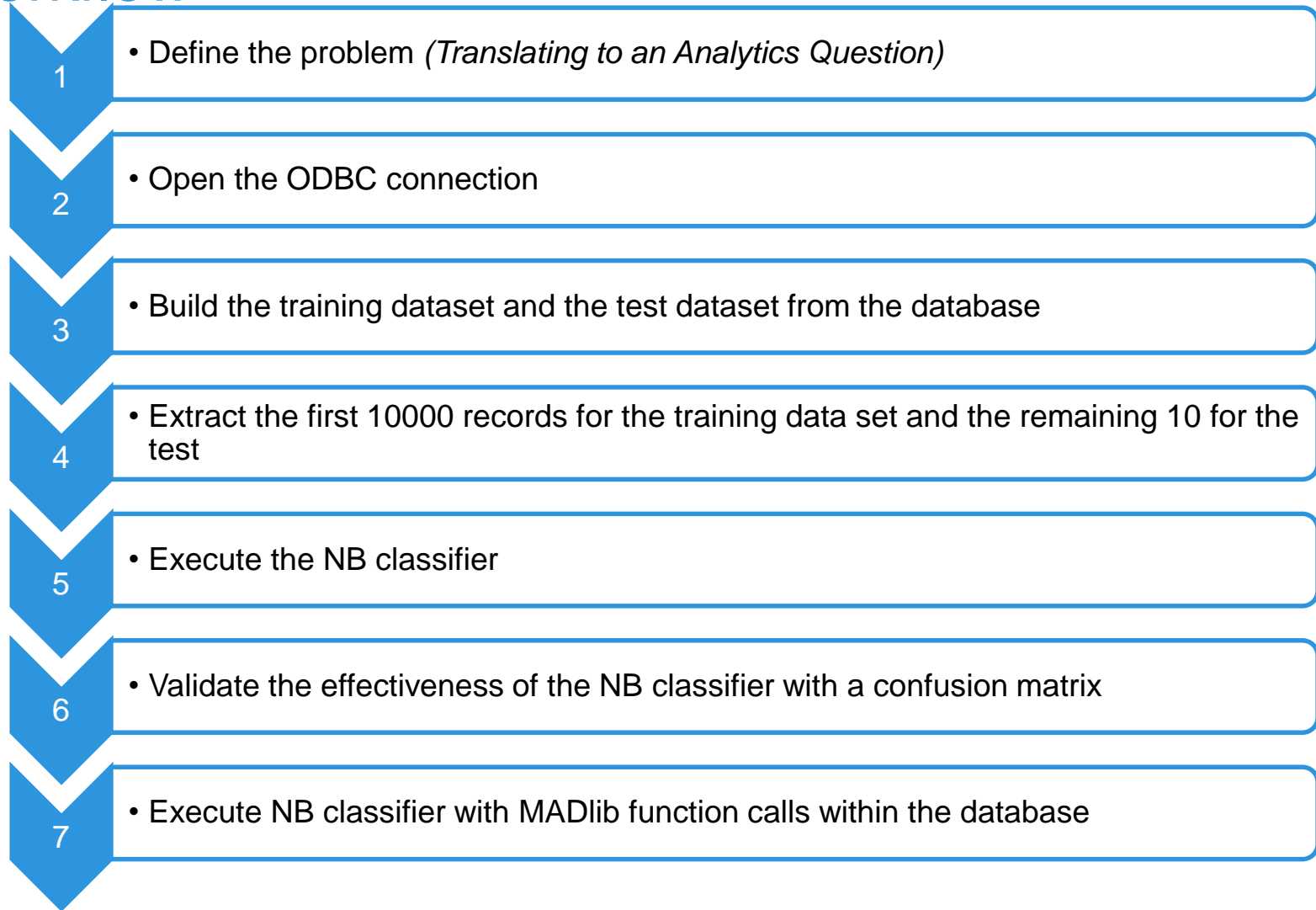
After completing the tasks in this lab you should be able to:

- Use R functions for Naïve Bayesian Classification
- Apply the requirements for generating appropriate training data
- Validate the effectiveness of the Naïve Bayesian Classifier with the big data

Lab Exercise 8: Naïve Bayesian Classifier Part1 - Workflow



Lab Exercise 8: Naïve Bayesian Classifier Part2 - Workflow



Thanks